Numerical Analysis (0/7)
Motivation & course plan

University of Luxembourg – 2024

Master in Mathematics

Hadrien Beriot

Outline

0.1 Introduction: motivations

0.2 Examples

0.3 Goal of the course & working plan

0.4 Plan of the lectures

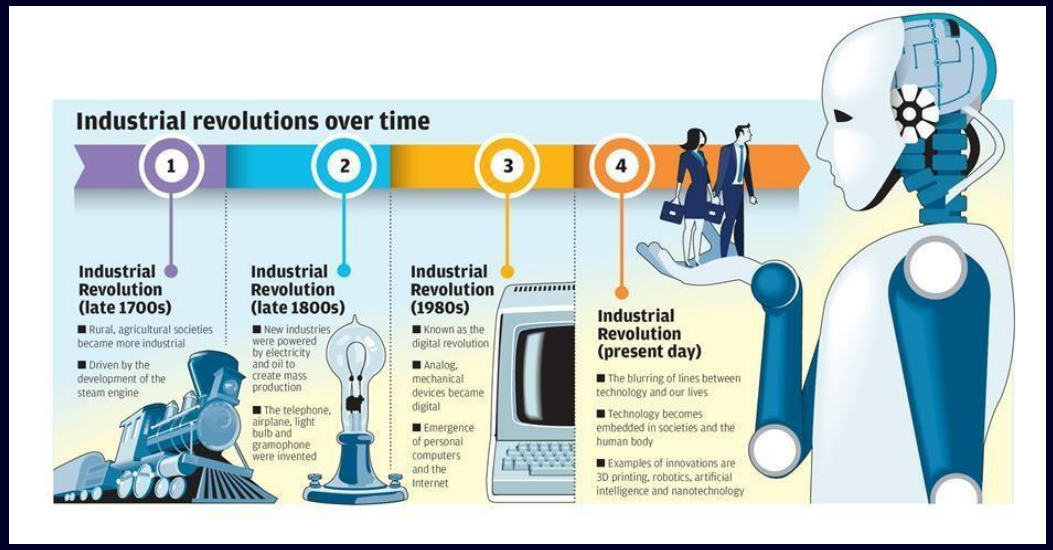
About me

- **R&D Manager at Siemens Industry Software NV. (Leuven)**
  - Engineering degree in France (2004)
  - PhD in Mechanical engineering (2008)
  - Joined Belgian company to work as a **developer** (2008)
  - **Researcher** in structural dynamics and acoustics (2013)
  - **Team leader** in Dynamics and acoustics research (2018)
  - R&D **Manager** for Mechanical applications (2023)

0.1 Introduction: motivation

Motivations

- Engineering and science problems are written in terms of mathematical equations
- Numerical simulation is crucial to solve them (chemistry, economy, finance, mechanics, physics, natural science, biology ...)
- Getting an explicit solution is not possible in real-life problems
- We calculate an approximation on a computer, obtained from a system with a finite number of degrees of freedom/unknowns (a number, a vector, or a matrix)
- Scientific Computing is fueling the 3rd and 4rd industrial revolutions



https://www.linkedin.com/pulse/crm-fourth-industrial-revolution-clint-oram/

Credits: Understanding How Computing Has_Changed the World' T. Misa

SIEMENS

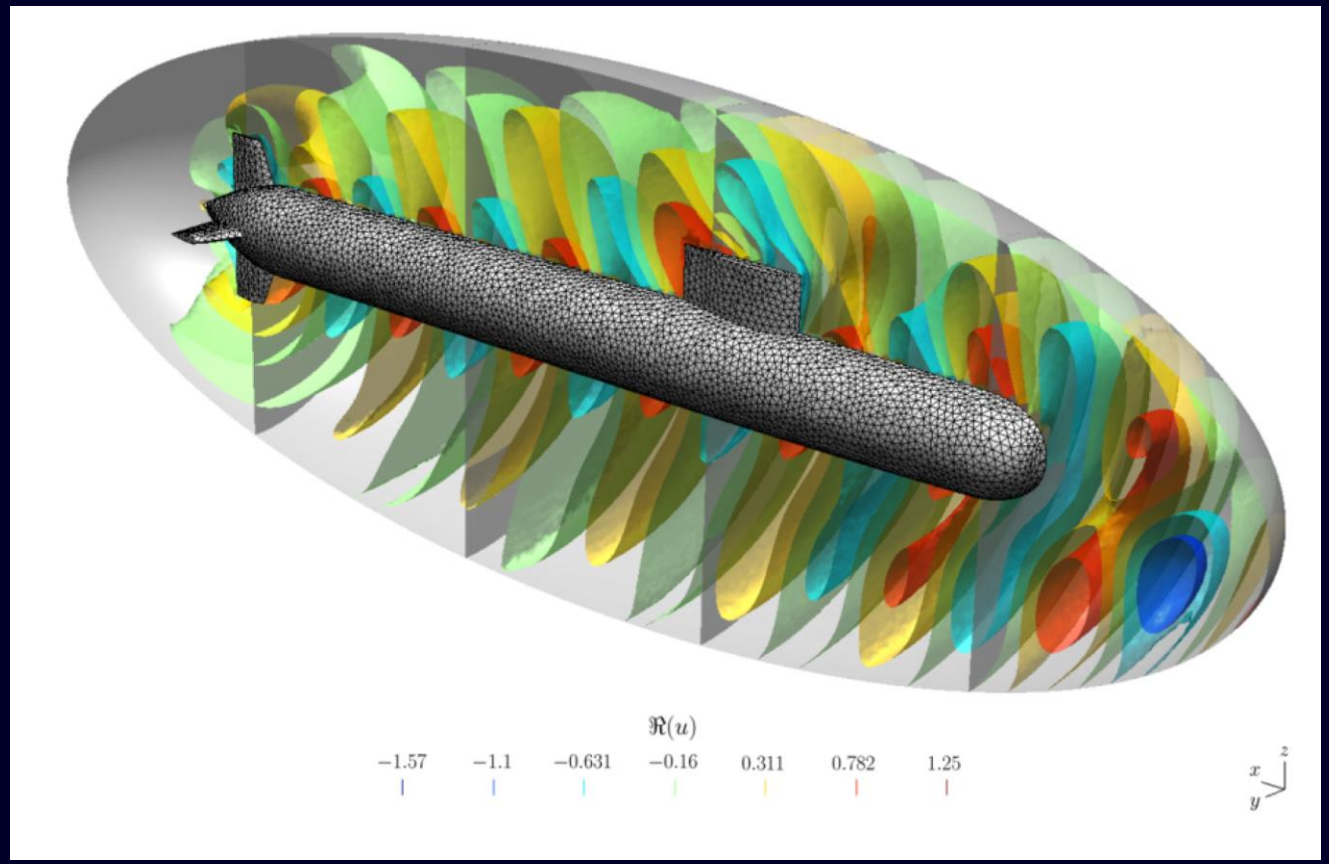
Motivations

- **Numerical analysis is multidisciplinary:**
  - design of methods for continuous mathematics (numerical analysts)
  - study of algorithms (computer scientists)
  - computer implementation and testing (software developers)

SIEMENS

0.2 Examples

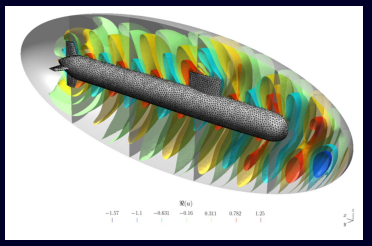
Example
plane wave scattered by a submarine



Credits: gmsh fem

SIEMENS

Weak formulation (Numerical solution of PDEs)

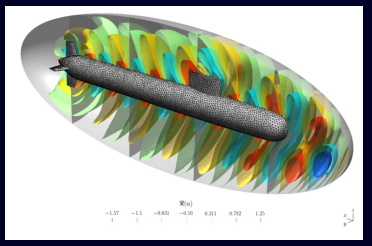


- What tools do we need to solve such a problem?
- First, a discrete representation of the geometry, here we have a 3D mesh (tetrahedrons),
- In practice, it is easier to solve a weak/variational formulation (finite elements)

$$\int \nabla u \cdot \nabla v - k^2 uv \, d\Omega + boundary\ terms = 0$$

- u: unknown (here the pressure perturbation)
- v: test-functions
- We solve the problem for example at the nodes/vertices of the mesh to obtain a vector $\boldsymbol{u}$.
- Between the nodes the solution is interpolated: Lagrange interpolation (Chapter 2)

Weak formulation (Numerical solution of PDEs)

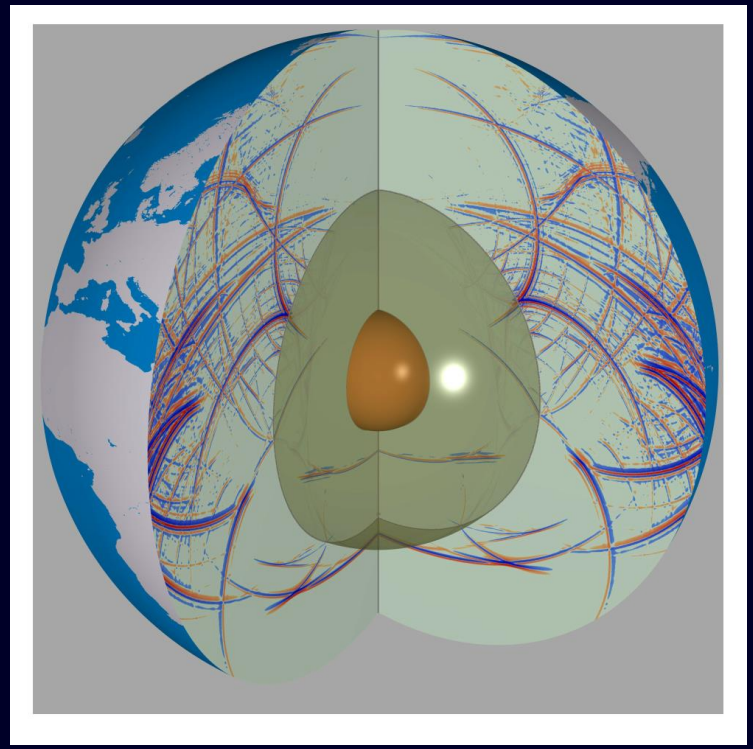


- **Solving the system**
  - Integrate accurately polynomials over elements (triangles, tetrahedrons): quadrature (Chapter 3)
  - In addition, computing the gradient needs numerical differentiation (Chapter 3)
  - The operator being linear, we need to solve a linear system: numerical linear algebra (Chapter 5)

- **Related problems**
  - Time-domain problem → dynamical system: resolution of ODEs (Chapter 4)
  - Nonlinear wavenumber dependency: k = k(u): solve nonlinear systems (Chapter 6)
  - A related problem concerns the eigenmodes of a vibrating structure: computation of eigenvalues & eigenfunctions (chapter 5)

Geophysics: discover the earth interior structure

- Sometimes we need to solve many problems for optimization purpose
  → minimize the error between measurements and simulations

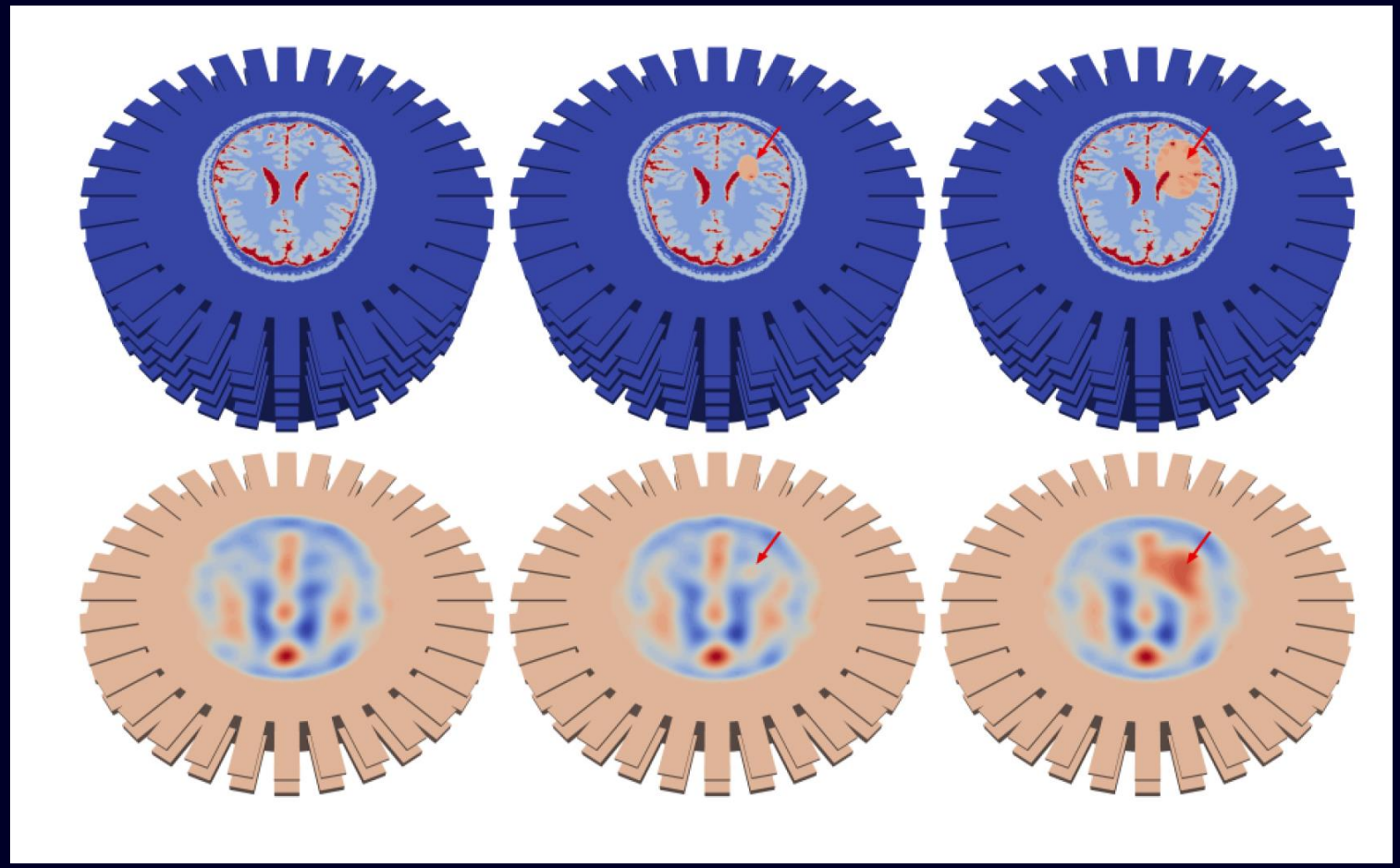


From LMU Geophysics - Munich

SIEMENS

Medicine: preventing cerebrovascular accidents

Cheap and rapid detection of brain defects

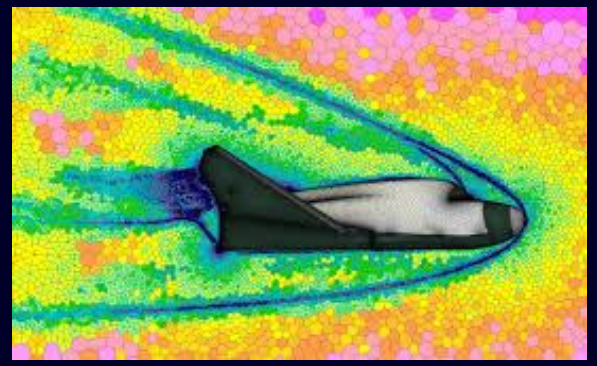


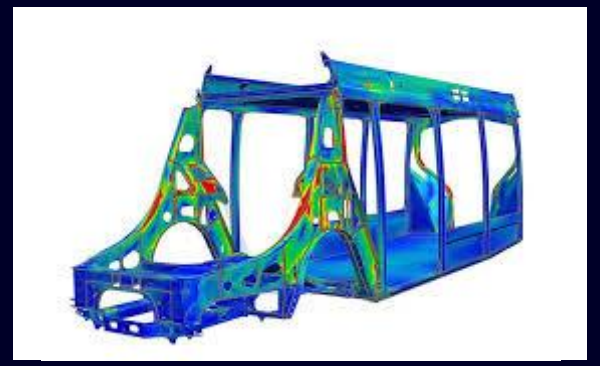
(2015 Bull-Joseph Fourier Prize)

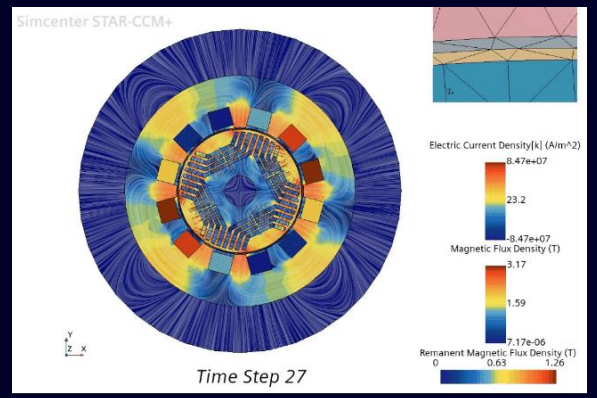
SIEMENS

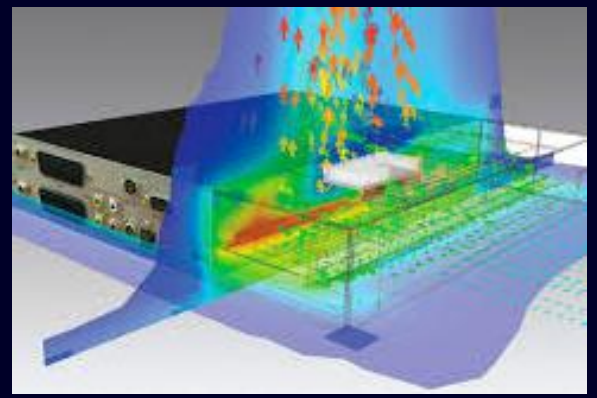
Last but not least

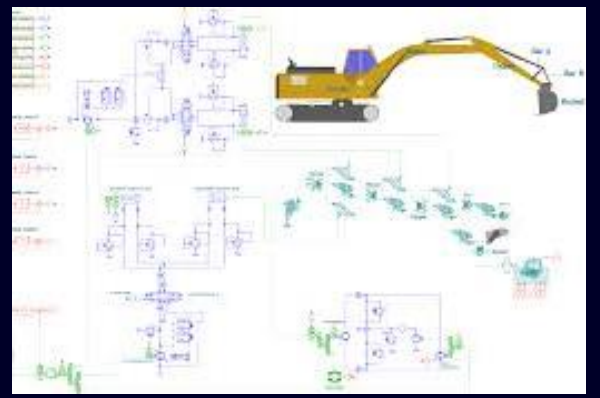
Electronics, Flow & thermal, Mechanical, Electro-mag, Space physics, Biology, Finance, etc.











SIEMENS

0.3 Goal of the course & working plan

Motivations

- **Main goals:**
  - To provide theoretical and practical elements related to numerical methods
  - Understanding how to go from the continuous to the discrete level (error estimates, algorithms, stability)
  - To be able to implement some methods (using Python)

- **Organization**
  - 1st half of the session: lecture, 2nd half: application/coding exercises with notebooks
  - The exercises are here to help you understand the theoretical notions
  - Grading: 50% assessment exercises + 50% group project

Good investment ⇒ success

SIEMENS

0.4 Plan of the lectures

Plan of the lectures

1. Python/notebook reminder, notion of error in numerical analysis
2. Interpolation - Approximation
3. Derivation - Integration (+ choose a project)
4. Numerical solution of ODEs
5. Linear systems
6. Nonlinear systems of equations
7. Project presentations